Identificación de flores Iris de tipo Versicolor, Setosa o Virginica

```
#-------------------------------------------------------------------------------
#   INSTALAMOS LIBRERÍAS
#-------------------------------------------------------------------------------

#-------------------------------------------------------------------------------
#   CARGAMOS LOS DATOS
#-------------------------------------------------------------------------------
from sklearn.datasets import load_iris
Dataset = load_iris()

# ------------------------------------------------------------------
# CORTE EN EL CONJUNTO DE DATOS
# ------------------------------------------------------------------
X = Dataset.data
y = Dataset.target

#-------------------------------------------------------------------
# VARIABLES DE ENTRENAMIENTO Y VALIDACIÓN
#-------------------------------------------------------------------
from sklearn.model_selection import train_test_split
X_train, X_val, y_train, y_val = train_test_split( X, y, random_state = 0, test_size = 0.20 )

#-------------------------------------------------------------------
#   CREAMOS EL MODELO
#-------------------------------------------------------------------
from sklearn.tree import DecisionTreeClassifier
arbol = DecisionTreeClassifier()

#-------------------------------------------------------------------
#   ENTRENAMOS EL MODELO
#-------------------------------------------------------------------
arbol.fit( X_train, y_train )

#-------------------------------------------------------------------
#   DIMENSIONAMOS EL ÁRBOL DE DECISIÓN
#-------------------------------------------------------------------
import matplotlib.pyplot as plt
plt.figure( figsize = ( 20, 10 ) )

from sklearn.tree import plot_tree
plot_tree(
            decision_tree = arbol,    # árbol de decisión regresor o clasificador
            #max_depth      = 5,    # int, none
            filled = True,
            feature_names  = Dataset.feature_names,   # list of str, none
            class_names    = Dataset.target_names    # list of str, bool
            #fontsize       = 7       # None, automático
            )
plt.show()
```

```
#-------------------------------------------------------------------------------
#   INSTALAMOS LIBRERÍAS
#-------------------------------------------------------------------------------

#-------------------------------------------------------------------------------
#   CARGAMOS LOS DATOS
#-------------------------------------------------------------------------------
import pandas as pd
df = pd.read_csv( "iris.csv" )

# ------------------------------------------------------------------
# CORTE EN EL CONJUNTO DE DATOS
# ------------------------------------------------------------------
X = df.iloc[:, :-1]
y = df.iloc[ :, -1]

#-------------------------------------------------------------------
# VARIABLES DE ENTRENAMIENTO Y VALIDACIÓN
#-------------------------------------------------------------------
from sklearn.model_selection import train_test_split
X_train, X_val, y_train, y_val = train_test_split( X, y, random_state = 0, test_size = 0.20 )

#-------------------------------------------------------------------
#   CREAMOS EL MODELO
#-------------------------------------------------------------------
from sklearn.tree import DecisionTreeClassifier
arbol = DecisionTreeClassifier()

#-------------------------------------------------------------------
#   ENTRENAMOS EL MODELO
#-------------------------------------------------------------------
arbol.fit( X_train, y_train )

#-------------------------------------------------------------------
#   DIMENSIONAMOS EL ÁRBOL DE DECISIÓN
#-------------------------------------------------------------------
import matplotlib.pyplot as plt
plt.figure( figsize = ( 20, 10 ) )

from sklearn.tree import plot_tree
plot_tree(
            decision_tree = arbol,    # árbol de decisión regresor o clasificador
            #max_depth      = 5,    # int, none
            filled = True,
            feature_names  = Dataset.feature_names,   # list of str, none
            class_names    = Dataset.target_names    # list of str, bool
            #fontsize       = 7       # None, automático
            )
plt.show()
```